Centralized Target Tracking with Propagation Delayed Measurements

Erdal Mehmetcik
Defense Systems Technologies Division
Sonar Systems Department
Aselsan Inc.
Ankara, Turkey
Email: emehmetcik@aselsan.com.tr

Umut Orguner
Department of Electrical and Electronics Engineering
Middle East Technical University
Ankara, Turkey
Email: umut@metu.edu.tr

Abstract—In this paper, we investigate the multisensor target tracking problem in environments where the signal propagation delays between the target and the sensors are not negligible. Based on a previously proposed efficient method to solve the implicit equations imposed by the target motion model and the physics rules governing the signal propagation delays, a particle filter is developed to perform multisensor target tracking while compensating for the propagation delays. As opposed to the existing approach in the literature, a centralized tracking architecture, which brings up significant challenges into the problem, is considered. The performance of the new approach is illustrated on an example scenario where a target with a coordinated-turn motion is to be tracked with two bearing-only sensors.

Index Terms—Centralized estimation, multiple sensors, propagation delay, state estimation, target tracking, particle filtering.

I. INTRODUCTION

Sensor systems used in target tracking [1], [2] observe the emitted (passive sensors) or reflected (active) energy from the target. Common examples of these sensors are radars, sonars, microphones, infrared sensors, cameras etc. Each of these sensors samples the emitted (or reflected) signal and depending on the propagation speed of the signal in the medium, there is a certain observation delay. In the case of radars, the propagation speed of the signal in the medium is the speed of light. As the emitted signal travels to the sensor, the change in target state within this short time duration can be neglected. However, the same assumption may not be as accurate while using acoustic sensors, since the propagation speed of acoustic waves can be comparable to the speed of the target to be tracked. Using acoustic sensors such as microphones or geophones is desirable in certain applications due to their low cost. In addition, for almost all underwater applications for target detection, the only feasible sensor selection is an acoustic sensor, the hydrophone, which converts the incoming acoustic pressure wave to electrical signals. In a variety of applications the propagation speed of the signal to be sampled by the sensors (speed of sound $\sim 344 m/s$ in air, $\sim 1500 m/s$ for water) is much less than the speed of light. Although for underwater applications, the speed of propagation ($\sim 1500 m/s$) is still much higher than a typical target speed (10 to 20 $m/s$), the propagation delays are still not negligible as the typical detection ranges are in the order of kilometers. As a result, when a target is observed at a specific time instant $t_k$ from an acoustic sensor, depending on the target speed, the measurement must be modeled as a function of the delayed state, $x_{k-}\Delta_{k}$, of the target rather than the state at the measurement time, $x_k$. With multiple sensors or possibly multiple arrays, even though the sensors collect measurements at synchronized time instants, the measurements will be completely asynchronized due to the propagation delays. The problem is also prominent in the following extreme case. Consider for instance the problem of tracking the position of a comet. In that case although the propagation speed of the observed signal is the speed of light, the observation still cannot be assumed as a function of the current state of the target, as the distance of the comet is in the order of thousands of kilometers and a considerable amount of time passes until the light waves (or particles) reach the sensor system. Also, the distance between the comet and the sensor changes over time with the motion of the comet and the resultant propagation delay is not a constant value and depends on the target dynamics together with the propagation speed.

In summary, depending on the propagation speed, the speed of the target under consideration and the distance between the sensor and the target, the propagation delays may not be negligible.

Before a general formulation of this problem is given in Section III, a simplified introduction to the problem to be solved is given next. Consider a tracking scenario where we have the perfect knowledge about the target state vector $x_{t_{k-1}}$ at time $t_{k-1}$. Each sensor acquires an observation which is a function the target state and generally all sensors are synchronized to produce their outputs at times $t_k$. The measurement system can acquire range, bearing or Doppler data. Because of the non-negligible signal propagation time, the obtained measurement from the $m^{th}$ sensor at time $t_k$ will be

$$y_k^m = h^m(x_{t_{k-1}} - \Delta_k^m) + \nu_k^m$$

(1)

where $h^m(.)$ is, in general, a non-linear function. The delay $\Delta_k^m$ can be described as a function of the distance between
the target and the $m^{th}$ sensor at time $t_k - Δ^m_k$ as follows.

$$Δ^m_k = d^m_k(x_{tk} - Δ^m_k)$$  \hspace{1cm} (2)

Also, using the target dynamics, the delayed state can be predicted as:

$$x_{tk} - Δ^m_k = f_{tk} - Δ^m_k, x_{tk-1}(x_{tk-1})$$  \hspace{1cm} (3)

The last two equations define an implicit constraint for the delay value $Δ^m_k$. As the number of sensors increase, additional constraints are added as well. This type of implicit and in general non-linear constraints, and the inclusion of these constraints to the state estimation procedure for the multisensor case are studied in this paper.

The defined problem was originally proposed and analyzed in [3] where a deterministic sampling based solution [4], [5], PDM (Propagation Delayed Measurement) filter, was proposed for the single sensor case. The method is later generalized to multiple sensor case in [6] using the largest ellipsoid algorithm (LEA) [7], [8] in a distributed tracking scheme. In [9] implicit delay constraints are utilized in particle filters [10], [11] for a single sensor case.

In this work, the idea will be extended to centralized multi sensor target tracking using particle filters.

II. SIMPLIFIED EXAMPLE

Leaving the general and formal problem formulation to Section III, we here make a simplified introduction to the problem. Consider, for instance, the case in Figure 1 where we observe a target whose scalar state $x_t$ is the position on the x-axis. The target has the known state $x_0 = 0$ at time $t_{k-1} = 0$ and moves with a known constant speed $v_T$ along the x-axis. At time $t_k$, a sound sensor $s_1$ positioned on the y-axis value $y_{s1}$ collects the bearing $φ^s_{1k}$ of the target (corresponding to time $t_k - Δ^1_k$). Then, the specific models corresponding to (2) and (3) would be

$$Δ^s_k = \frac{1}{v_s} \sqrt{y^2_{s1} + x^2_{tk} - Δ^s_k},$$  \hspace{1cm} (4)

$$x_{tk} - Δ^s_k = v_T(t_k - Δ^s_k)$$  \hspace{1cm} (5)

respectively, where $v_s$ is the speed of sound. These two equations have a solution $Δ^s_k > 0$ satisfying

$$v^2_T(t_k - Δ^s_k)^2 + y^2_{s1} = (v_s Δ^s_k)^2$$  \hspace{1cm} (6)

as shown in Figure 1. Now, instead of solving the parabolic equation for $Δ^s_k$, one can define a recursion for $Δ^s_k$ with the initial value e.g. $Δ^s_k(0) = 0$ by substituting (5) into (4) to get

$$Δ^s_k(m + 1) = \frac{1}{v_s} \sqrt{y^2_{s1} + [v_T(t_k - Δ^s_k(m))]^2}$$  \hspace{1cm} (7)

which can be shown to converge to the positive root of (6) if $v_T < v_s$.

Just as in the case of this simple example, in a more general setting, (2) and (3) together define an implicit equation for $Δ^s_k$ (like (6)) whose solution can be obtained with iterative techniques similar to (7). This type of implicit and in general nonlinear constraints and their inclusion into the estimation process were examined in the work [3]. In this simplified scenario, we have neglected three sources of uncertainty:

- The initial state $x_{tk-1}$ is actually random.
- A process noise term must be added into the simplified description (3).
- The propagation time itself through (2) might be uncertain due to possible reasons such as the uncertain position of the sensor.

The PDM filter of [3] covers all the uncertainties mentioned above for the single sensor case and is based on including the delay $Δ^s_k$ in the state vector while processing the observation taken at time $t_k$. Since the measurements observe the delayed state values $x_{tk} - Δ^s_k$, the augmented state is formed as $[x_{tk} - Δ^s_k, Δ^s_k]^T$ in the PDM filter.

In the multiple sensor case, even if the sensors are synchronized to acquire measurements at a specific time $t_k$, each sensor observes the target at a different time instant $t_k - Δ^s_k$ due to different sensor positions (see the case of two sensors illustrated in Figure 1). While making centralized multisensor estimation, a direct extension of the PDM filter requires all delayed states $x_{tk} - Δ^s_k$ and the corresponding delays $Δ^s_k$ to be included into an augmented state which would significantly increase the dimension of the state-space. Due to this reason, the study [6] preferred to handle the multisensor case only in a distributed setting where each local agent calculates its own estimate by running a local PDM filter with its own augmented state. In this work, to be able to deal with the multisensor estimation problem in a centralized setting, we propose to overcome this difficulty by defining the overall state as the target state only at the measurement times, i.e. $x_{tk}$, and then retrodict the propagation delayed states $x_{tk} - Δ^s_k$ and use the corresponding delays $Δ^s_k$ to make the measurement update. The details of the proposed approach will be given in Section IV. A formal definition of the problem is given in the next section.
### III. Problem Formulation

The following discrete-time nonlinear state space model is considered:

\[ x_{tk+1} = f(x_{tk}, t_k) + w_{tk+1, tk} \]  

where \( x_{tk} \in \mathbb{R}^{n_x} \) is the state sequence with initial distribution \( x_{t_0} \sim p_0(x_{t_0}) \). The state dynamics given in (8) is a discretized version of the corresponding continuous time dynamics given as

\[ \dot{x}_t = f(x_t) + w_t \]  

In (8), \( t_k \in \mathbb{R} \) is an arbitrary time value and \( f(t_{k+1}, t_k) \) is the state transition function transforming \( x_{tk} \) to \( x_{tk+1} \) according to the continuous time dynamics \( f(\cdot) \). \( \{w_{tk+1, tk}\} \in \mathbb{R}^{n_w} \) is a white noise sequence with distribution \( w_{tk+1, tk} \sim p_w \).

It is assumed that the transformation in (8) is invertible. That is to say the following operations are possible:

\[ x_t = f(x_t), \quad w_t \]

The reason for restricting the state transition models to invertible ones will be clear in the next section, since invertibility is required to construct the proposed algorithm.

The propagation delayed measurements \( y^S_k \) are collected with the sensors shown by \( s = 1, \ldots, S \) which are given as

\[ y^s_k = h^s_k(x_{tk} - \Delta^s_k) + \nu^s_k \]  

where \( t_k \) is the time at which the measurement is collected at the sensor side, \( h^s_k(\cdot) \) is a general nonlinear function, \( \Delta^s_k \) is the propagation delay in the measurement \( y^s_k \) and \( \nu^s_k \in \mathbb{R}^{n_y} \) is a white measurement noise sequence independent from the process noise with distribution \( \nu^s_k \sim p_w(\cdot) \). The knowledge about the propagation delay \( \Delta^s_k \) is available in the form of an implicit equation (called \( c^s_k \)) as follows:

\[ c^s_k : \Delta^s_k = d^s_k(x_{tk} - \Delta^s_k) + \tau^s_k \]  

where \( d^s_k(\cdot) \) is a general non-linear function of the delayed state value \( x_{tk} - \Delta^s_k \) and \( \tau^s_k \in \mathbb{R} \) is a white noise sequence independent from the process and measurement noise with distribution \( \tau^s_k \sim p_w(\cdot) \). In the case of a passive acoustic sensor, the delay expression is given as

\[ \Delta^s_k = \frac{||p(t^s_k - \Delta^s_k) - p^s||}{v_s} + \tau^s_k \]  

where \( p(t^s_k - \Delta^s_k) \) is the position of the target at time \( t^s_k - \Delta^s_k \) and \( p^s \) is the position of the sensor \( s \) and \( v_s \) is the speed of sound in the propagation medium. The noise term \( \tau^s_k \) represents the unpredictable effects in the transmission of the acoustic wave through the environment or the possible uncertainty in the sensor position.

The aim is to find a possibly approximate expression for the density \( p(x_{tk}|y_{0:tk}; c^s_{0:tk}) \).

### IV. Centralized multisensor PDM-PF

As explained in [9], it is possible to recursively find the delayed state from which the measurement originated (i.e. \( x_{tk} - \Delta^s_k \)) such that the delay constraint in (13) holds. Although the method in [9] was derived to predict the future state values using the state transition model, we will use the method to propagate the particles backwards in time to find the propagation delayed states \( x_{tk} - \Delta^s_k \). Hence, as previously stated, the process model needs to be invertible. The reason for propagating the states backward in time is to handle the measurement update stage for multiple sensors by using only a single set of particles that represent the density of \( x_{tk} \).

The backward propagation method can be briefly summarized as follows:

- Start with the following initial delay value.

\[ \Delta_k^u(0) = d(x_{tk}) + \tau \]  

- Calculate the delayed state and the new delay values until convergence

\[ x_{tk} - \Delta^u_k(j) = f(x_{tk} - \Delta^u_k(j), t_k) + \tilde{w} \]

\[ \Delta_k^u(j + 1) = d(x_{tk} - \Delta^u_k(j)) + \tau \]

In [9] it is shown that the sequence \( \Delta^u_k(j) \) converges exponentially to the fixed point \( \Delta(\infty) \) satisfying:

\[ \Delta(\infty) = d(x_{tk} - \Delta(\infty)) + \tau \]

For simplicity of the notation in the following sections, the above recursive algorithm will be referred to as

\[ [x_{tk} - \Delta, \Delta] = \text{recursion}(x_{tk}, p^s) \]  

where \( p^s \) is the position of the \( s \)-th sensor.

In [6] and [9] the augmented target state vector includes the delayed kinematic state vector \( x_{tk} - \Delta \) which is directly related to the measurements by (12). Although such a state definition is well suited for the implementation of state estimation algorithms for a single sensor, it is hard to generalize the approach to the multisensor case. This is because of the fact that each sensor might have a different position leading to different propagation delays and therefore for each sensor a different kinematic state vector \( x_{tk} - \Delta_k \) should be included in the augmented state. With such an approach, the size of the augmented state would increase linearly with the number of sensors which would lead to inefficiencies and even infeasibilities in the case of a particle filter due to the curse of dimensionality. Another difficulty caused by the approach of [6] and [9] is that in order to calculate an output estimate for \( x_{tk} \), an additional step of prediction (from the propagation delayed state values \( x_{tk} - \Delta_k \)) is required.

In this work, to overcome difficulties of the aforementioned approach, the state vector is defined as the target state at time \( \{t_k\} \), i.e., \( x_{tk} \), which is the state vector of the target at the times the measurements are collected. Hence the particle set \( \{x_{tk}^{(i)}\}_{i=1}^N \) and corresponding weights \( \{\pi_k^{(i)}\}_{i=1}^N \) are used to represent the state density. The propagation delayed target states \( x_{tk} - \Delta^u_k \) which are directly related to the
measurements (12) are then obtained using retrodiction (i.e., a prediction backwards in time), at each time step, from the particles \( \{x_{t_k}^{(i)}\}_{i=1}^N \). Using these retrodicted states, the measurement update can be done using a standard particle filter measurement update. The main advantage of defining the state vector in this new way, is the ability to directly obtain a state estimate for time \( t_k \) using the current particles without an extra prediction step. More importantly, the state-space dimension does not grow as the number of sensors increases with the new approach. As expected, the number of retrodicted states is equal to the number of sensors and hence the computational load will scale linearly with the number of sensors.

The details of the prediction and measurement updates of the particle filter to obtain the new summary statistics \( \{\hat{x}_{t_k}^{(i)}\}_{i=1}^N \) and \( \{\pi_k^{(i)}\}_{i=1}^N \) are given as follows.

- Assume that there are \( S \)-many stationary sensors and the position vector of the \( k \)-th sensor is given as \( p_k \). Suppose that the summary statistics at time \( t_{k-1} \) is available, i.e. \( \{\hat{x}_{t_{k-1}}^{(i)}\}_{i=1}^N \) and \( \{\pi_k^{(i)}\}_{i=1}^N \) are given.

**Prediction update**:

- Propagate each particle to the next time instant using the state transition model.
  \[
  \hat{x}_{t_k}^{(i)} = f_{t_k,t_{k-1}}(x_{t_{k-1}}^{(i)}) + u_{t_k,t_{k-1}}^{(i)}
  \]  
  (20)
  where \( u_{t_k,t_{k-1}}^{(i)} \sim p_{t_k,t_{k-1}}^{(i)} \).

- Using the recursion (19), generate the propagation delayed particles for each sensor, by retrodiction.
  \[
  \{\hat{x}_{t_k-\Delta_k}^{(i),m}\}_{i=1}^N = \text{recursion}(\hat{x}_{t_k}^{(i)}, p_m) \]  
  (21)
  where \( \Delta_k^{(i),m} \) is the delay associated with the \( m \)-th sensor for the \( i \)-th particle.

**Measurement update**:

- Using the predicted particles \( \{\hat{x}_{t_k}^{(i)}\}_{i=1}^N \) and their weights \( \{\pi_k^{(i)}\}_{i=1}^N \), the measurement updated density \( p(x_{t_k}|y_k^{1:S}, c_k^{1:S}) \) can be obtained by updating the weights as
  \[
  \pi_k^{(i)} \propto p(y_k^{1:S}|\hat{x}_{t_k}^{(i)})\pi_k^{(i) - 1} \]  
  (22)

The main issue with the update method described above is the evaluation of the likelihood value \( p(y_k^{1:S}|\hat{x}_{t_k}^{(i)}) \) in the measurement update equation. Using the Chapman-Kolmogorov equation, the likelihood can be written as

\[
p(y_k^{1:S}|\hat{x}_{t_k}^{(i)}) = \int p(y_k^{1:S}|X_{t_k-\Delta})p(X_{t_k-\Delta} | \hat{x}_{t_k}^{(i)})dX_{t_k-\Delta}
\]  
(23)

where the vector \( X_{t_k-\Delta} \) defined as

\[
X_{t_k-\Delta} \triangleq \left[ x_{t_k-\Delta}^T \cdots x_{t_k-\Delta}^S \right]^T
\]  
(24)
is composed of the propagation delayed state vectors for the sensors. Assuming that the process noise, \( u_{t_k,t_{k-1}} \), and measurement noise, \( \nu_k \), terms are independent Gaussian processes (i.e., \( u_{t_k,t_{k-1}} \sim N(0, Q_{t_k,t_{k-1}}) \), \( \nu_k \sim N(0, R_k) \)) and using (12) and (8), the expression (23) can be written as

\[
p(y_k^{1:S}|\hat{x}_{t_k}^{(i)}) = \int N(Y_k; H_k(X_{t_k-\Delta}), R)
\]
\[
\times N(X_{t_k-\Delta}; X_{t_k-\Delta}, Q_{t_k-\Delta,t_k}) \)dX_{t_k-\Delta}.
\]  
(25)

where

\[
X_{t_k-\Delta} \triangleq \left[\begin{array}{c}
\left(x_{t_k-\Delta}^{(i)}ight)^T \\
\vdots \\
\left(x_{t_k-\Delta}^S\right)^T
\end{array}\right]
\]  
(26)
\[
Q_{t_k-\Delta,t_k} \triangleq \left[\begin{array}{c}
Q^{11,(i)} \\
\vdots \\
Q^{SS,(i)}
\end{array}\right]
\]  
(27)
\[
Q^{m,(i)} \triangleq Q_{\max}(t_k-\Delta_k^{(i),m}, t_k-\Delta_k^{(i)}, t_k)
\]  
(28)
\[
H_k(X_{t_k-\Delta}) \triangleq \left[\begin{array}{c}
h_k^1(x_{t_k-\Delta}^{(i)}) \\
\vdots \\
h_k^S(x_{t_k-\Delta}^{(i)})
\end{array}\right]^T
\]  
(29)
\[
y_k \triangleq \left[\begin{array}{c}
y_k^1 \\
\vdots \\
y_k^S
\end{array}\right]^T
\]  
(30)
\[
R \triangleq \text{blkdiag}[R_1, ..., R_S]
\]  
(31)

Since the functions \( h_k^s(\cdot) \) are in general nonlinear, the integral (25) cannot be computed analytically. At this point, we propose the following approaches to approximate the integral (25).

- Small process noise assumption: The first approximation to calculate (25) is to assume that \( Q_{t_k-\Delta,t_k} \approx 0 \) which simply leads to the approximation
  \[
  N(X_{t_k-\Delta}; X_{t_k-\Delta}, Q_{t_k-\Delta,t_k}) \approx \delta_{X_{t_k-\Delta}}(X_{t_k-\Delta})
  \]  
(32)
where \( \delta(\cdot) \) denotes a Dirac delta function.
Substituting (32) into the integral (25), we obtain

\[
p(y_k^{1:S}|\hat{x}_{t_k}^{(i)}) \approx N\left(Y_k; H_k(X_{t_k-\Delta}^{(i)}), R\right)
\]  
(33)
\[
= \prod_{s=1}^S \int N\left(y_k^s; h_k^s(x_{t_k-\Delta}^{(i)}), R_s\right)
\]  
(34)

- Linearization: Another approach to approximate the integral (25) is to linearize \( H_k(\cdot) \) around the mean value \( X_{t_k-\Delta}^{(i)} \) as
  \[
  H_k(X_{t_k-\Delta}) \approx H_k(X_{t_k-\Delta}^{(i)}) + \frac{\partial H_k}{\partial X_{t_k-\Delta}}|_{X_{t_k-\Delta}=X_{t_k-\Delta}^{(i)}}
  \times \left(X_{t_k-\Delta} - X_{t_k-\Delta}^{(i)}\right)
  \]  
(35)
\[
= H_k(X_{t_k-\Delta}^{(i)}) + H_k \left(X_{t_k-\Delta}^{(i)} \right)
\]  
(36)
where

\[ H_k = \frac{\partial H_k}{\partial x_{tk-\Delta}} \bigg|_{x_{tk-\Delta}=x_{tk-\Delta}^{(i)}}. \] (37)

This approximation yields the result

\[
p(y_k^{1:S} | x_{tk}^{(i)}) \approx N \left( Y_k; H_k \left( x_{tk-\Delta}^{(i)} \right), H_k Q_{tk-\Delta,t_k} H_k^T + R \right). \] (38)

- Unscented transform: The last idea we present to approximate the integral (25) is to use Unscented transform. In this approach we generate the sigma points \( \{ x_{tk-\Delta}^{(j)}; w_{tk-\Delta}^{(j)} \}_{j=1}^{2nx} \) that represent the density \( N \left( X_{tk-\Delta}; X_{tk-\Delta}^{(i)}, Q_{tk-\Delta,t_k} \right) \). Then the integral (25) can be approximated as

\[
p(y_k^{1:S} | x_{tk}^{(i)}) \approx \sum_{j=1}^{2nx} w_{tk-\Delta}^{(j)} N \left( Y_k; H_k (x_{tk-\Delta}^{(j)}), R \right) \] (39)

Hence, we have three different methods to carry out the measurement update stage given in (22). After the measurement update stage, the estimate can be calculated as

\[
\hat{x}_{tk} = \sum_{i=1}^{N} \pi_k^{(i)} \hat{x}_{tk}^{(i)}
\] (40)

We give a brief summary of one step of the proposed particle filter for the sake of clarity below.

**Centralized Multsensor PDM-PF Algorithm:**

- Suppose the summary statistics \( \{ x_{tk-1}^{(i)}, \pi_{k-1}^{(i)} \}_{i=1}^{N} \) is available at time \( t_{k-1} \).
- Resample particles:

\[
\left\{ x_{tk-1}^{(i)}, \pi_{k-1}^{(i)} \right\}_{i=1}^{N} = \text{resample} \left( \left\{ x_{tk-1}^{(i)}, \pi_{k-1}^{(i)} \right\}_{i=1}^{N} \right)
\] (41)

- Predict the particles at the next time instant using the process model:

\[
\hat{x}_{tk}^{(i)} = f_{tk-1} \left( x_{tk-1}^{(i)} \right) + w_{tk-1,t_{k-1}}^{(i)}
\] (42)

where \( w_{tk-1,t_{k-1}}^{(i)} \sim \mathcal{N}(0, Q_{tk-1}) \) and \( i = 1, ..., N \) (N: Number of particles).
- Using the recursion in (19), predict the propagation delayed states for all sensors:

\[
\hat{x}_{tk-\Delta_j} = \text{recursion} \left( \hat{x}_{tk}^{(i)} \right)
\] (43)

for \( i = 1, ..., N \) and \( j = 1, ..., S \) (S: Number of sensors).
- Using either one of three approaches explained in the previous section, update the particle weights:
  - (i) Negligible process noise assumption:

\[
\pi_k^{(i)} = \pi_{k-1}^{(i)} \prod_{s=1}^{S} N \left( y_{k_s}^{(i)}; h_k \left( x_{tk-\Delta_k}^{(i)} \right), R^s \right). \]

\] (44)
  - (ii) Linearization:

\[
\pi_k^{(i)} = \pi_{k-1}^{(i)} \times N \left( Y_k; H_k \left( X_{tk-\Delta}^{(i)} \right), H_k Q_{tk-\Delta,t_k} H_k^T + R \right).
\] (45)
  - (iii) Unscented transform:

\[
\pi_k^{(i)} = \pi_{k-1}^{(i)} \sum_{j=1}^{2nx} w_{tk-\Delta}^{(j)} N \left( Y_k; H_k (x_{tk-\Delta}^{(j)}), R \right)
\] (46)

- Normalize particle weights:

\[
\pi_k^{(i)} = \pi_k^{(i)} / \sum_{j=1}^{N} \pi_j^{(i)}
\] (47)

- Calculate the state estimate and covariance as:

\[
P_k = \sum_{i=1}^{N} \pi_k^{(i)} \left[ \left( \hat{x}_{tk}^{(i)} - \hat{x}_{tk} \right) \left( \hat{x}_{tk}^{(i)} - \hat{x}_{tk} \right)^T \right]
\] (49)

V. SIMULATION STUDY

In this section, the performance of the proposed algorithm (all 3 versions) is going to be compared on a simulated multistatic bearings only target tracking scenario. The following acronyms will be used to indicate proposed versions of the algorithm and the compared algorithms in the literature:

- MSPDMPF: Centralized multisensor propagation delayed measurement particle filter (proposed in this work), that assumes negligible correlation between back propagated states.
- MSPFMPF-EKF: Centralized multisensor propagation delayed measurement particle filter (proposed in this work), that carries out the measurement update stage with linearization approximation (Since the approximation resembles extended Kalman filter, EKF suffix is added to the acronym).
- MSPDMPF-UKF: Centralized multisensor propagation delayed measurement particle filter (proposed in this work), that carries out the measurement update stage with unscented transform. (Since the approximation resembles unscented Kalman filter, UKF suffix is added to the acronym).
- EKF-LEA: Distributed EKF algorithm (used in [6] for comparison). This filter completely ignores the propagation delays. Estimates are fused using the largest ellipsoid algorithm.
- EKFD-LEA: Distributed EKF algorithm, proposed in [6] that compansates the delays. Estimates are fused using the largest ellipsoid algorithm.
The simulated two dimensional bearings only scenario is shown in Figure-2. The simulated target performs a coordinated turn with the turn rate $-0.12\text{rads/sec}$ and an initial speed of $60\text{m/sec}(=216\text{km/hour})$ in the y-direction starting at the $[-500m, 800m]$ position for 45 seconds. There are two stationary sensors located at $[0m, 0m]$ and $[-750m, 750m]$. The sensors acquire noisy bearing measurements that are corrupted by independent Gaussian noises with zero mean and a standard deviation of $0.05\text{rads} \approx 3\text{degs}$. The sampling period for the measurements is $T_s = 1\text{sec}$. Both sensors start to acquire bearing data starting at $t_0 = 4\text{secs}$ and a total of 42 measurements are obtained in the time interval $[4\text{secs}, 45\text{secs}]$. At each time instant the particle filters are updated with the measurements. The local estimates of the distributed EKF-LEA and EKFD-LEA algorithms are fused at each time instant as well.

The target motion is modeled as a discretized coordinated turn model, [12], with an unknown turn rate augmented to the state vector. Hence the state vector is used as, $[p_k, p'_k, v_k, v'_k, \omega_k]^T$, where $p$, $v$ and $\omega$ denote position, speed and turn rate, respectively.

$$x_{k+1} = \begin{bmatrix} 1 & 0 & \sin(w_k T_{k+1}) & 0 & -\cos(w_k T_{k+1}) \\ 0 & 1 & 1-\cos(w_k T_{k+1}) & \sin(w_k T_{k+1}) & w_k \\ 0 & 0 & \cos(w_k T_{k+1}) & -\sin(w_k T_{k+1}) & 0 \\ 0 & 0 & \sin(w_k T_{k+1}) & \cos(w_k T_{k+1}) & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \\ w_{k+1} \end{bmatrix} + w_{k+1} \quad (50)$$

where $T_{k+1} = t_{k+1} - t_k$ and $w_k$ is Gaussian noise term with zero mean with covariance $Q_k$ given as

$$Q_k = \begin{bmatrix} \frac{\sigma_w^2 T_k^3}{3} & 0 & \frac{\sigma_w^2 T_k^2}{2} & 0 & 0 \\ 0 & \frac{\sigma_w^2 T_k^3}{3} & 0 & \frac{\sigma_w^2 T_k^2}{2} & 0 \\ \frac{\sigma_w^2 T_k^2}{2} & 0 & \sigma_w^2 T_k & 0 & 0 \\ 0 & \frac{\sigma_w^2 T_k^2}{2} & 0 & \sigma_w^2 T_k & 0 \\ 0 & 0 & 0 & 0 & \sigma_w^2 T_k \end{bmatrix} \quad (51)$$

In the simulations, the standard deviations for the turn rate and speed are $\sigma_w = 0.01\text{rad/s}^2$ and $\sigma_v = 1\text{m/s}^2$ respectively. The propagation speed is $344\text{m/sec}$. The initial particle set is generated from the distribution $N(\mu, \text{diag}(100^2, 100^2, 10^2, 10^2, 0.05^2))$ whose mean value is also a random variable $\mu \sim N(x_1, \text{diag}(100^2, 100^2, 10^2, 10^2, 0.05^2))$. The initial states of the EKF-LEA and EKFD-LEA are generated from the weighted mean and covariance estimates of the initial particle sets of the particle filters. A total of 1000 Monte-Carlo run is carried out by re-generating the measurement noises and the initial particles. Particle filters are run with 1000 particles. The particles whose speed estimate exceed the propagation speed are assigned a zero weight as well in all the implementations.

The RMS position and velocity errors are given in Figure 3 and Figure 4 respectively. The performance of the centralized algorithms turned out to be superior in all time instants in terms of the position error. In velocity estimation, the centralized algorithm performances are surprisingly worse than those of the distributed algorithms in the first 30 seconds of the scenario. On the other hand, the final velocity errors obtained by the centralized approaches become smaller than those of distributed ones at the last 10 seconds. When we compare different centralized approaches, the EKF and UKF extensions of the proposed centralized algorithm (MSPDMPF) does not seem to provide a significant improvement. This might be
attributed to the fact that the target makes a perfect coordinated turn in the scenario which corresponds to the case of no process noise. Therefore the small process noise assumption actually holds for this specific example. The evaluation of the performance of EKF and UKF extensions in a more realistic scenario where process noise is not negligible is left as a future work.

VI. CONCLUSION

In this paper a centralized multisensor particle filtering framework is presented to handle propagation delayed measurements. The upside of the presented method is that the dimension of the state space does not increase with the increasing number of sensors as would be the case in a direct extension of the PDM filter. The performance of the proposed algorithm is shown to be superior to its counterparts which work in a distributed tracking framework. The future work would be the evaluation of the framework which has different measurement update extensions in more realistic scenarios. Another interesting idea would be the investigation of the centralized multisensor setting with deterministic sampling based methodologies in [3] instead of the particle filters.

ACKNOWLEDGMENT

The authors would like to thank Assoc. Prof. Dr. Çağatay Candan of the Electrical and Electronics Engineering Department of Middle East Technical University for the valuable discussions which improved the quality of the paper. The second author thanks Prof. Fredrik Gustafsson of Linköping University, Sweden for his earlier contributions to their common work on the problem considered in this work.

REFERENCES


